Init data

```
In[∘]:= r1 = 1000; (*Ом*)
r2 = 1500; (*Ом*)
r3 = 2700; (*Ом*)
c = 10 000 * 10^-12; (*Ф*)

Δr1 = 0.05; (*%*)
Δr2 = 0.01; (*%*)
Δr3 = 0.01; (*%*)
Δc = 0.1; (*%*)

δ = 0.01;

t := 2 * r1 * c * Log[1 + 2 r2 / r3];
                  натуральный логарифм
Tanalytic[R1_, R2_, R3_, Cp_] := 2 * R1 * Cp * Log[1 + 2 R2 / R3];
                                               натуральный логарифм

ConvertArg[name_, var_] := Row[{name, var}, "="];
                           ряд
ConvertDeviation[name_, var_] := Row[{name, Row[{var * 100, "%"}]}, "="];
                                 ряд          ряд
```

Sensitivity coefficient

Derivative

Analytic

```
D[Tanalytic[R1, R2, R3, Cp], R1]
дифференцировать
D[Tanalytic[R1, R2, R3, Cp], R2]
дифференцировать
D[Tanalytic[R1, R2, R3, Cp], R3]
дифференцировать
D[Tanalytic[R1, R2, R3, Cp], Cp]
дифференцировать
```

Out[∘]= $2\,Cp\,\mathrm{Log}\left[1+\frac{2\,R2}{R3}\right]$

Out[∘]= $\frac{4\,Cp\,R1}{\left(1+\frac{2\,R2}{R3}\right)R3}$

Out[∘]= $-\frac{4\,Cp\,R1\,R2}{\left(1+\frac{2\,R2}{R3}\right)R3^2}$

Out[∘]= $2\,R1\,\mathrm{Log}\left[1+\frac{2\,R2}{R3}\right]$

Numeric

In[∘]:=
```
CalcD := Module[{ar1 = 0, ar2 = 0, ar3 = 0, ac = 0},
        программный модуль
  ar1 = N[(Tanalytic[(r1 + r1 * δ), r2, r3, c] - Tanalytic[(r1 - r1 * δ), r2, r3, c]) /
        численное приближение
      (2 (r1 * δ))];
  ar2 = N[(Tanalytic[r1, (r2 + r2 * δ), r3, c] - Tanalytic[r1, (r2 - r2 * δ), r3, c]) /
        численное приближение
      (2 (r2 * δ))];
  ar3 = N[(Tanalytic[r1, r2, (r3 + r3 * δ), c] - Tanalytic[r1, r2, (r3 - r3 * δ), c]) /
        численное приближение
      (2 (r3 * δ))];
  ac = N[(Tanalytic[r1, r2, r3, (c + c * δ)] - Tanalytic[r1, r2, r3, (c - c * δ)]) /
       численное приближение
      (2 (c * δ))];
  Return[{ar1, ar2, ar3, ac}]
  вернуть управление
 ]
```

Absolute sensitivity

Coefficients

In[∘]:=
```
Module[{ar1 = 0, ar2 = 0, ar3 = 0, ac = 0, list = CalcD},
программный модуль
 ar1 = list[[1]];
 ar2 = list[[2]];
 ar3 = list[[3]];
 ac = list[[4]];
 Row[{ConvertArg["A_R1", ar1], ConvertArg["A_R2", ar2],
 ряд
   ConvertArg["A_R3", ar3], ConvertArg["A_C", ac]}, "\n"]
]
```

Out[∘]=
A_R1 = 1.49443×10^{-8}
A_R2 = 7.01761×10^{-9}
A_R3 = -3.89886×10^{-9}
A_C = 1494.43

Equation

In[•]:=
```
Module[{r1 = ΔR1, r2 = ΔR2, r3 = ΔR3, c = ΔC, coeffs = CalcD, left, right},
 right = Row[{ Row[{coeffs[[1]], r1}], Row[{coeffs[[2]], r2}],
    Row[{coeffs[[3]], r3}], Row[{coeffs[[4]], c}]}, "+"];
 left = ΔT;
 Row[{left, right}, "="]]
```
программный модуль; ряд

Out[•]= $\Delta T = 1.49443 \times 10^{-8} \Delta R1 + 7.01761 \times 10^{-9} \Delta R2 + -3.89886 \times 10^{-9} \Delta R3 + 1494.43 \Delta C$

Relative sensitivity

Coefficients

In[•]:=
```
CalcB := Module[{br1 = 0, br2 = 0, br3 = 0, bc = 0, list = CalcD},
  br1 = list[[1]] * r1 / t;
  br2 = list[[2]] * r2 / t;
  br3 = list[[3]] * r3 / t;
  bc = list[[4]] * c / t;
  Return[{br1, br2, br3, bc}]
 ]
```
программный модуль; вернуть управление

In[•]:=
```
Module[{br1 = 0, br2 = 0, br3 = 0, bc = 0, list = CalcB},
 ConvertArg[name_, var_] := Row[{name, var}, "="];
 br1 = list[[1]];
 br2 = list[[2]];
 br3 = list[[3]];
 bc = list[[4]];
 Row[{ConvertArg["B_R1", br1], ConvertArg["B_R2", br2],
   ConvertArg["B_R3", br3], ConvertArg["B_C", bc]}, "\n"]
]
```
программный модуль; ряд

Out[•]=
```
B_R1 = 1.
B_R2 = 0.704377
B_R3 = -0.70441
B_C = 1.
```

Equation

In[∘]:=
```
Module[
{r1 = ΔR1 / R1, r2 = ΔR2 / R2, r3 = ΔR3 / R3, c = ΔC / C, coeffs = CalcB, left, right},
 right = Row[{ Row[{coeffs[[1]], r1}], Row[{coeffs[[2]], r2}],
    Row[{coeffs[[3]], r3}], Row[{coeffs[[4]], c}]}, "+"];
 left = ΔT / T;
 Row[{left, right}, "="]]
```
программный модуль; генерируемая константа; ряд

Out[∘]= $\frac{\Delta T}{T} = 1.\frac{\Delta R1}{R1} + 0.704377\frac{\Delta R2}{R2} + -0.70441\frac{\Delta R3}{R3} + 1.\frac{\Delta C}{C}$

Maximum deviations

Worst case method

Absolute

In[∘]:=
```
Module[{r1 = ΔR1, r2 = ΔR2, r3 = ΔR3, c = ΔC, coeffs = CalcD, left, right},
 CrSum[index_] := Abs[coeffs[[index]]];
 right = Row[{Row[{CrSum[1], r1}],
    Row[{CrSum[2], r2}], Row[{CrSum[3], r3}], Row[{CrSum[4], c}]}, "+"];
 left = ΔT;
 Row[{left, right}, "="]]
```
программный модуль; абсолютное значение; ряд

Out[∘]= $\Delta T = 1.49443\times 10^{-8}\Delta R1 + 7.01761\times 10^{-9}\Delta R2 + 3.89886\times 10^{-9}\Delta R3 + 1494.43\Delta C$

Relative

In[∘]:=
```
Module[{cfs = CalcB, max, min},
 max = cfs[[1]] * Δr1 + cfs[[2]] * Δr2 + cfs[[3]] * Δr3 + cfs[[4]] * Δc;
 min = cfs[[1]] * (-Δr1) + cfs[[2]] * (-Δr2) + cfs[[3]] * (-Δr3) + cfs[[4]] * (-Δc);
 Row[{ConvertDeviation["ΔT/T max", max], ConvertDeviation["ΔT/T min", min]}, "\n"]]
```
программный модуль; ряд

Out[∘]= ΔT/T max = 15.%
ΔT/T min = -15.%

Probabilistic method

Absolute

In[ ]:=
```
Module[{cfs = CalcB, γ = 1, δ},
программный модуль
 δ = γ * Sqrt[Power[cfs[[1]] * Δr1 * γ, 2] + Power[cfs[[2]] * Δr2 * γ, 2] +
         ква⋯ степень                          степень
     Power[cfs[[3]] * Δr3 * γ, 2] + Power[cfs[[4]] * Δc * γ, 2]];
     степень                        степень
 Row[{ConvertDeviation["ΔT/T max", δ]}, "\n"]]
 ряд
```

Out[ ]=
```
ΔT/T max = 11.2246%
```

Relative

```
Module[{cfs = CalcB},
программный модуль
 max = cfs[[1]] * Δr1 + cfs[[2]] * Δr2 + cfs[[3]] * Δr3 + cfs[[4]] * Δc;
 min = cfs[[1]] * (-Δr1) + cfs[[2]] * (-Δr2) + cfs[[3]] * (-Δr3) + cfs[[4]] * (-Δc);
 Row[{ConvertDeviation["ΔT/T max", max], ConvertDeviation["ΔT/T min", min]}, "\n"]]
 ряд
```

Out[ ]=
```
ΔT/T max = 15.%
ΔT/T min = -15.%
```